

How to Allocate Resources For Features Acquisition?

Oran Richman Department of Electrical Engineering Technion Haifa, Israel

`roran@tx.technion.ac.il`

Shie Mannor Department of Electrical Engineering Technion Haifa, Israel

`shie@ee.technion.ac.il`

July 12, 2016

Abstract

We study classification problems where features are corrupted by noise and where the magnitude of the noise in each feature is influenced by the resources allocated to its acquisition. This is the case, for example, when multiple sensors share a common resource (power, bandwidth, attention, etc.). We develop a method for computing the optimal resource allocation for a variety of scenarios and derive theoretical bounds concerning the benefit that may arise by non-uniform allocation. We further demonstrate the effectiveness of the developed method in simulations.

1 Introduction

Most machine learning settings take feature vectors as input. These features are often acquired using some process resulting in less than optimal data quality. In many situations, the data quality depends on the resources allocated for the data acquisition process. Examples of possible resources are sample rate, total sample time, CPU allocated to some costly pre-processing, and transmitted power.

Several approaches have been proposed in order to deal with some uncertainty in learning schemas (for example [20, 27]). In many cases, however, one does not merely deal with existing uncertainty, but can sometimes “shape” the uncertainty to meet one’s needs. This is often the case when several sensors share a common resource. For example, mobile applications use sensors that share power, CPU and bandwidth. Each of those resources can be divided between sensors according to the designer wish. Another example is the design of a system with fixed budget (money wise), each type of sensor incorporated can have a variety of qualities (with a price tag to match). Which sensor is “worth” investing in?

In this work we explore the following problem: Several sensors that share a common resource acquire inputs that will be used for classification. What is the best way to divide the resource between the sensors? The resources allocated for each sensor affect the quality of the data it collects. We wish to maximize classification performance by correctly allocating the available resources. We emphasize that different resource allocation schemes may result in different optimal classifiers. This coupling increases the complexity of the problem.

We present a framework for “uncertainty management”: This framework formulates the presented problem as an optimization problem. The direct formulation, however, is not easily solvable so we derive an equivalent solvable problems for various scenarios. We further bound the benefit that may arise from optimally allocating the resources. Based on the results presented we devise an algorithm for deriving the optimal resource allocation and present some simulation results that show the potential benefits.

An application domain of such an approach is that of sensor management (see [6]), where mostly state-estimation problems have been investigated. Among the most studied applications is the real-time allocation of radar resources (for example [25]). However, other applications such as multi sensor management [26] have also been studied. One more emerging application is the use of services like Mechanical Turk in order to extract features (for example subjective features regarding an image or a text). The more averaging performed, the more accurate the features are. However, not all features require the same accuracy.

In our model, collected features are corrupted by some disturbance. We explore two types of disturbances: stochastic and adversarial. A stochastic disturbance corresponds to common situations where features are corrupted by some, typically additive, noise. An adversarial disturbance concerns the worst possible deterministic loss maximizing noise corresponding to “worst-case” scenarios.

We assume that special effort is made so that the training data are of the highest quality. During the test phase, however, resources are limited and should be allocated sparingly. This is often the case in applications where the number of samples to be classified is larger by several orders of magnitude than the training set size. This work focuses on methods for controlling uncertainty in problems of binary classification with real valued features. We consider support vector machines (SVM) style classification [7] due to its many beneficial properties (for example [22] and [28]). However, our method can be easily adapted to a wide variety of learning schemes.

We further explore a second scenario in which we assume that the training data are noisy while during the test phase data quality is superb. This can occur for a number of reasons. One example is some difficulty to gather information in the learning phase which do not exist in the test-phase. For example, patients may be more willing to conduct a CT scan when some serious illness is suspected but convincing them to perform one for the sake of experimentation require the use of less radiation therefore more noise [4]. Another example is when the learning data-set is “sensitized” by artificiality adding noise in order to comply with privacy issues. Scenarios in which noise arise in both training and testing phase can be accommodated by a combination of the methods presented.

In most of the paper we assume that the relation between the resources to be allocated and the disturbance is known. This scenario is quite reasonable, examples include influence of sampling rate on temporal features, sampling time on spectral features, power on channel error rate in communication and many more. However, since there are also cases where this relation is unknown we introduce an algorithm that is *completely* data-driven. We do not

assume Gaussian noise. However, in many areas of control and signal processing Gaussian noise is used to model sensors noise. For that reason the examples given consider Gaussian noise.

Related works. The problem of resource allocation between sensors has been investigated in several disciplines and from several perspectives. Most works come from an adaptive control perspective. Almost half a century ago, Meier [14] defined a setting where sensors parameters can be controlled. The control perspective has been studied extensively since, mostly for the special case of sensor switching, namely dynamically choosing one sensor from several available ones; see [2] and many others. In contrast with those works we are dealing with classification problem. The existence of some decision boundary makes the problem more involved and the control theory framework inadequate. In addition, this line of research generally assumes full knowledge of the underline model, an assumption we would like to avoid.

In [3] the authors considered the problem of finding an optimal least-squares linear regressor as well as noise parameters of a static estimation problem when the underlying model is known. They explore the spacial case of estimating a scalar using square loss. A mild extension to this spacial case is given in [19]. We generally follow the same approach, although our problem definition is more general. We fortunately have the privilege of enjoying a later rich body of research concerning dealing with known uncertainty in learning scenarios (e.g., [20, 27]).

Classification problems in this context were considered by trying to maximize some measure of information in the data. In this setting one tries to optimize some information measures like sample conditional entropy or the Kullback-Leibler (KL) divergence (for example [8]). Such methods lead to an elegant solution but are heuristic and ignore knowledge about the desired utility function, so that some information “quantity” is optimized instead of the relevance to classification.

Resource efficient learning is a growing field of research in recent years. Most research

is focused on dynamic acquisition of features where different features are acquired for different samples. Multiple models were proposed including trees [30], cascades [23] and Markov decision processes [5]. Our work explores the situation where features are acquired simultaneously and not sequentially. Some work had also considered introducing resource awareness into the classifier learning process. This is usually done using some greedy process where features are added to a classifier until the resource budget run out [16, 29]. Similar methods which treat the learning scheme as “black-box” are wrapper feature selection [10]. Some work had explored similar issues when resources are scarce in the learning phase instead of the testing phase [12, 15]. While our work shares a similar motivation with those fields, our decision space is continuous and not discrete. We are inspired by problems in which sensors use a physical resource which need to be allocated (time, power, bandwidth, etc.). Existing methods cannot support such problems. In addition, the use of a continuous decision space circumvents the need to solve complex combinatorial problems and allows the use of various tools from optimization theory.

Another setting which had been explored is on-line learning in the presence of noise. An algorithm for on-line learning from noisy data is presented in [4]. We improve the algorithm presented there by allowing on-line control of features quality and show that learning can be done more efficiently.

Contributions. The contributions of this paper are:

- We develop a framework for considering feature acquisition quality as a resource allocation problem in classification.
- We derive algorithms for optimal resource allocation and optimal classification for a variety of scenarios.
- We analyse the performance gain that can be achieved.
- We demonstrate the benefit that can arise from using those methods in simulation.

The structure of this paper is as following: Section 2 introduces the framework of uncertainty management and provides a method for determining the optimal resource allocation for stochastic disturbances. Section 3 explores the case of adversarial noise. The results presented in those sections characterize the optimal allocation for a wide array of problems. Section 4 proposes an algorithm for the scenario where the disturbance characteristics is unknown and gives a theoretical guarantee on its regret. Section 5 explores the case where the training set is noisy and provides an efficient algorithm for the special case of linear classifier with Gaussian noise and square loss. Section 6 presents some simulations that demonstrate the feasibility of the results and Section 7 concludes with some final thoughts. Proofs for all of the theorems in this paper can be found in the appendix.

2 Uncertainty allocation: Stochastic disturbances

This section explores the case in which the disturbance is stochastic. We assume that M samples $(x, y) \in (\mathbb{R}^d, \{-1, 1\})$ are generated from some joint distribution (i.i.d.). Denote by X_{ij} the j 'th feature of sample i . Each X_{ij} is measured with some disturbance δ_{ij} . The disturbance is generated from a distribution with some vector of parameters (resources) $r = (r_1, \dots, r_d)$. Denote the resulting vector of disturbances in sample i as δ_i . We follow the empirical risk minimization framework [24]. Let $L(h, r)$ be the cost incurred when the disturbance is generated using resource vector r . That is.

$$L(h, r) = \frac{1}{M} \sum_{i=1}^M \mathbb{E}_{\delta} (l(h, X_i + \delta_i, Y_i)).$$

Our objective is to optimize *both* the resource vector (r_1, \dots, r_d) and the classifier $h(x)$ such that $L(h, r)$ is minimized.

For simplicity, we focus our attention on the spacial case of linear classifiers. However, the framework presented in this paper can be easily extended to other families of classifiers. Also, we assume that the noise is independent between samples, namely that each δ_{ij} is

generated i.i.d. using a distribution with parameter r_j . We note that $L(h, r)$ can also be written as $L(h, r) = \tilde{L}(h, \sigma(h, r))$ where $\sigma(h, r) \in \mathbb{R}$. The variable $\sigma(h, r)$ is a measure of the noise influence on the cost function. For example, for linear classifiers it is often the standard deviation of the noise in the axis perpendicular to the decision boundary. This is helpful since many loss function may be defined this way. We give details of such an example below (Example 1).

For a linear classifier we define,

$$L(w, b, r) = \frac{1}{M} \sum_{i=1}^M \mathbb{E}_{\delta} (l(Y_i, (w^T(X_i + \delta_i) + b))).$$

Assume that $\sigma(\cdot, \cdot, r)$ is a convex function in r , positive and strictly decreasing in each element for $r_j > 0$. Also, assume that $\tilde{L}(w, b, \sigma)$ is strictly increasing and convex in σ . We refer to a loss function that satisfies these assumptions as an *acceptable loss function*. Those assumption can be informally interpreted as assuming that more resources provide better accuracy and that increasing performance provide diminishing return.

The problem can now be stated as:

$$\begin{aligned} \min_{r, w, b} \quad & L(w, b, r) \triangleq \tilde{L}(w, b, \sigma(w, b, r)) \\ \text{s.t.} \quad & \sum_{i=1}^d r_i \leq R \\ & r_j \geq 0 \quad \forall j. \end{aligned} \tag{1}$$

Example 1 Consider the case in which δ_{ij} is Gaussian with zero mean and standard deviation $\sigma_j(r_j)$, where $\sigma_j(r_j)$ is a convex strictly decreasing function. Assume also that $l(x, y, w, b) = l(w^\top x + b, y)$. In this case, σ is the standard deviation of the distance from the decision boundary, namely, $\sigma(w, b, r) = \sqrt{\sum_{i=0}^d w_i^2 \sigma_i(r_i)^2}$.

Now, there are two natural loss functions we can explore: hinge loss and square loss. For the hinge loss $l(x, y, w, b) = \max(0, 1 - y(w^\top x + b))$. In such case $L(w, b, r)$ can be calculated

directly:

$$L(w, b, r) = \frac{1}{M} \sum_{i=1}^M \left(\frac{1}{\sqrt{\pi}\sigma(w, b, r)} \int_{-\infty}^1 (1-z) e^{-\frac{(Y_i(w^\top X_i + b) - z)^2}{2\sigma(r)^2}} dz \right).$$

For the square loss $l(x, y, w, b) \triangleq (w^\top x + b - y)^2$. In this case a simple calculation shows that the overall loss is:

$$L(w, b, r) = \sigma(w, b, r)^2 + \frac{1}{M} \sum_{i=1}^M (Y_i - (w^\top X_i + b))^2. \quad (2)$$

Similarly, one can use other loss functions and obtain a numerical if not exact expressions.

The following theorem characterizes the optimal resource allocation for problem (1). According to Theorem 1 the resource allocation depends only on $\sigma(w, b, r)$ such that one can derive the optimal resource allocation even without knowing $L(w, b, r)$. The proof of the theorem and all other proofs appear in the appendix.

Theorem 1. Suppose that $L(w, b, \sigma)$ is an acceptable loss function. For the optimal solution (w, b, r) of problem (1) there exists $\lambda > 0$ such that $\sum_{i=1}^d r_i = R$, and for every i it holds that

$$\begin{aligned} r_i &= 0 & \text{if } -\frac{\partial \sigma}{\partial r_i}(w, 0) < \lambda \\ -\frac{\partial \sigma}{\partial r_i} &= \lambda & \text{else} \end{aligned} \quad (3)$$

Using Theorem 1 and greedy search over λ problem (1) can be solved.

2.1 Examples

We now outline a few examples of optimal allocation of resources for different relations between the resources and the noise variance. While all of the examples relate to zero mean Gaussian noise, Theorem 1 is general and can be applied for other distributions as long as their variance is finite.

Example 2 - Standard deviation proportional to inverse of resource We explore the scenario in which the standard deviation is proportional to the inverse of the resources allocated. Namely, $\sigma_i(r_i) = \frac{1}{r_i}$. This is the case, for example, when the resource is the sampling rate and the features measured are timing of various events. In this case:

$$r_i = \frac{Rw_i^{\frac{2}{3}}}{\sum_{j=1}^d w_j^{\frac{2}{3}}}.$$

Example 3: Variance proportional to inverse of resources A popular relation between resources and noise is when the variance is proportional to the inverse of resources allocated. Namely, $\sigma_i(r_i) = 1/\sqrt{r_i}$. This is the case in many situations including: power in active sensors, duration of sampling for spectral features and number of measurements taken when averaging (for example if features are extracted using a Mechanical Turk). In this case, the optimal allocation can be easily computed to be:

$$\hat{r}_i = \frac{R|w_i|}{|w|_1}. \quad (4)$$

Corollary 1. In the case of square-loss and uniform allocation of resources $r_i = R/d$, it follows that $\sigma(r)^2 \propto |w|_2^2$.

Corollary 2. When applying optimal allocation of resources according to (4) it results that $\sigma(r)^2 = |w|_1^2/R$.

Interestingly, the optimization problem derived for square-loss (2) with uniform allocation of resources is equivalent to the optimization problem derived when performing ridge regression. Similarly, using optimal allocation of resources is similar to performing lasso regularization. This support claims that using lasso regularization produce classifiers which are more robust to noise than other regularization techniques [27]

Since for the square-loss optimizing (2) is equivalent to performing lasso, one can use complicity bounds derived for this case. It is known that a bound on the error resulting

from lasso regularization $|w|_1 < B$ is increasing in B [9]. Since B is decreasing with $1/R$, it is increasing with R . This surprisingly implies that *less resources require less examples to learn*. This can be explained in the following manner: with less resources there is more noise in the decision making phase, the larger the noise the less impact small changes in the classifier makes (in the limit, there are no resources and therefore the noise is infinite and there is nothing to learn).

Example 4: Quantization noise It is known that rounding quantization noise can be treated as Gaussian with standard deviation of $\frac{1}{12}LSB$ where the LSB is the accuracy of the least significant bit [21]. Consider a scenario in which we would like to maintain the number of bits used to represent all features under some threshold R . We will first disregard the fact that r must be an integer and derive the solution for $\sigma_i(r) = 2^{-(r_i)}$ while $r_i \geq 1$ for all i . The solution of (1) for any fixed w will be

$$\begin{aligned} r_i &= \begin{cases} 1 & \log |w_i| < \lambda \\ 1 + \log |w_i| + \frac{1}{|C|}(R - d - \sum_{i \in C} \log |w_i|) & \log |w_i| \geq \lambda \end{cases} \\ \lambda &= \frac{1}{|C|}(\sum_{i \in C} \log |w_i|) - R + d \\ C &= \{i | \log |w_i| \geq \lambda\}. \end{aligned}$$

Notice that we still need to transform r_i into integers. This can be done by “searching” in the vicinity of the optimal vector r .

2.2 Performance analysis

To gain some insight about the expected benefit of using this method we explore the special case of square loss with $\sigma_i(r_i) = 1/\sqrt{r_i}$. Observe that L is decreasing in R . From Corollary 1 and 2 we know that finding an optimal w for uniform allocation is equivalent to performing ridge regression while optimizing (2) is equivalent to performing lasso. We ask the following question: for the same expected loss how much resources can we save by using the method

presented? In order to answer this question we start by fixing w and analyze the expected loss for different resources allocations.

For every admissible (w, b) denote by $R_{unif}(w, l)$ the resource budget that holds $L(w, b, r_{unif}) = l$ when $r_{unif} = (R/d, \dots, R/d)$. Also, denote by $R_{opt}(w, l)$ the resource budget that holds $L(w, b, r_{opt}(w)) = l$ when $r_{opt}(w) = (R|w_1|/|w|_1, \dots, R|w_d|/|w|_1)$. The following result bounds the ratio between resources required for achieving the same loss.

Theorem 2. For every w and for $l(x, y, w, b) = (w^\top x + b - y)^2$ it holds that $\frac{R_{unif}(w, l)}{R_{opt}(w, l)} = \frac{d|w|_2^2}{|w|_1^2}$.

The proof can be found in the appendix.

Denote by w_{opt} the optimal classifier when resources are allocated optimally and by w_{unif} the optimal classifier when resources are allocated uniformly. The next corollary follows directly from Theorem 2; it bounds the total benefit that can arise from the joint optimization of both resource allocation and classifier. It holds since $L(w_{unif}, r_{unif}) \leq L(w_{opt}, r_{unif})$ and $L(w_{opt}, r_{opt}(w_{opt})) \leq L(w_{unif}, r_{opt}(w_{unif}))$.

Corollary 3. For every w and for $l(x, y, w, b) = (w^\top x + b - y)^2$ it holds that

$$\frac{d|w_{unif}|_2^2}{|w_{unif}|_1^2} \leq \frac{R_{unif}(w_{unif}, l)}{R_{opt}(w_{opt}, l)} \leq \frac{d|w_{opt}|_2^2}{|w_{opt}|_1^2}.$$

It is clear from Corollary 3 that in cases where some features hold little information (small coefficients in the classifier) the benefit of optimized resource allocation can be very large. It should be noted that in extreme cases this is equivalent to using feature selection (meaning, choosing which features should be allocated zero resources). However, in many cases even when considering only relevant features the variance of their influence is significant. In such cases our method provides considerable benefit.

3 Adversarial disturbance

We now consider the case where the disturbance is adversarial. Several models for adversarial disturbance have been considered in the literature, we will adopt the model from [27]. Formally, consider some samples $\{(X_i, Y_i)\}_{i=1}^M$ where $X_i \in \mathcal{X} \subset \mathbb{R}^d$ and $Y_i \in \{-1, 1\}$. We only have access to some corrupted version of this set $\{(X_i + \delta_i, Y_i)\}_{i=1}^M$. The disturbances δ_i are determined by an adversary, however the adversary can only affect samples in a certain way. Formally, the vector $\delta = (\delta_1, \dots, \delta_M)$ is in a set defined by : $\mathcal{N}(\mathcal{N}_0) \triangleq \{(\alpha_1 \delta_1, \dots, \alpha_M \delta_M) | \delta_i \in \mathcal{N}_0 \text{ for } i = 1, \dots, M, \sum_{j=1}^M \alpha_j = 1\}$, where \mathcal{N}_0 is some symmetric uncertainty set that contains the origin.

In our setting, we wish to optimize both the classifier parameters w, b and the shape of \mathcal{N}_0 under the constraint of available power (or budget) for the adversary. The main difference here from other works (see [27] and follow-ups) is that we can *optimize* over \mathcal{N}_0 out of a family of sets (set of sets). Such a family can be, for example, the set of ellipsoid sets while maintaining some constant fixed resource budget.

$$\mathcal{N}^{set} = \left\{ \mathcal{N}_0 = \left\{ x \mid \sum_{i=1}^d \left(\frac{x_i}{\sigma_i(r_i)} \right)^2 \leq 1 \right\}, \quad \sum_{i=0}^d r_i = R \right\}.$$

Formally, the problem is optimizing

$$\inf_{\mathcal{N}_0 \in \mathcal{N}^{set}} \sup_{\delta \in \mathcal{N}(\mathcal{N}_0)} \min_{w, b} L(w, b, X + \delta, Y) \quad (5)$$

for some \mathcal{N}^{set} that defines the problem. We will focus our attention on the hinge loss, $L \triangleq \sum_{i=1}^M \max(0, (1 - y_i(< w, X_i > + b)))$.

For hinge loss the following result is given in [27]:

Lemma 1. (Xu et al. 2009 [27]) Assume $\{X_i, Y_i\}_{i=1}^M$ are non-separable then the following min-max problem

$$\min_{w, b} \sup_{\delta \in \mathcal{N}} \sum_{i=1}^M \max(0, (1 - y_i(< w, X_i > + b)))$$

is equivalent to the following optimization problem

$$\begin{aligned}
& \min_{w,b,\xi} \sup_{\delta \in \mathcal{N}_0} (w^T \delta) + \sum_{i=1}^M \xi_i \\
& s.t. \\
& \xi_i \geq 1 - y_i(w^T x_i + b), \quad \xi_i \geq 0, \quad i = 1 \dots, M.
\end{aligned}$$

We use this result in order to derive the following theorem:

Theorem 3. Consider the solution $(\hat{N}_0, \hat{w}, \hat{b})$ for the problem

$$\inf_{\mathcal{N}_0 \in \mathcal{N}^{set}} \min_{w,b} \sup_{(\delta_1, \dots, \delta_M) \in \mathcal{N}} L(w, b, X + \delta_i, Y). \quad (6)$$

Then the solution satisfies

$$\hat{N}_0 \in \arg \min_{\mathcal{N}_0 \in \mathcal{N}^{set}} \sup_{\delta \in \mathcal{N}_0} (w^T \delta).$$

Theorem 3 is analogous to Theorem 1 and allows to optimize resource allocation in the adversarial setting.

Example 4 Gaussian noise is a popular modelling choice in many domains. We wish to find some constraint which will create in the adversarial setting an effect that reassembles Gaussian noise. For this purpose we use an ellipsoid uncertainty set. Instead of assuming Gaussian noise we bound the uncertainty to a fixed width of standard deviations. Consider the model presented in [27] with an ellipsoid uncertainty set namely, $\mathcal{N}_0 = \{x \mid \sum_{i=1}^d (x_i / \sigma_i(r_i))^2 \leq 1\}$. The function $\sigma_i(r_i)$ can be any of the former examples.

Now, under non separability assumption the solution of the problem

$$\min_r \min_{w,b} \sup_{(\delta_1, \dots, \delta_M) \in \mathcal{N}(r)} \sum_{i=1}^M \max(1 - y_i(w^T (x + \delta_i) + b), 0), \quad s.t. \quad \sum_{i=0}^d r_i = R$$

satisfies,

$$w_i^2 \sigma_i(r_i) \frac{d\sigma_i(r_i)}{dr_i} = \lambda, \quad \sum_{i=1}^d r_i = R.$$

Fixing σ_i allows the derivation of w, b by solving the conic optimization problem

$$\begin{aligned} \min_{w, b, \xi} & \sqrt{\sum_{i=1}^d w_i^2 \sigma_i^2} + \sum_{i=1}^M \xi_i \\ \text{s.t.} & \\ & \xi_i \geq 1 - y_i(w^T x_i + b), \quad \xi_i \geq 0, \quad i = 1 \dots, M. \end{aligned}$$

Theorems 1 and 3 allow to solve either (1) or (5) using alternating optimization.

Moreover, in the special case where $\sigma_i(r_i) = 1/\sqrt{r_i}$ the optimal allocation of resources is analogous to lasso regression:

$$\sqrt{\sum_{i=1}^d w_i^2 \sigma_i^2} = \frac{|w|_1}{\sqrt{R}}, \quad r_i = \frac{R|w_i|}{|w|_1} \text{ for } i = 1, 2, \dots, d. \quad (7)$$

4 Unknown stochastic disturbance

In this section we consider the case of stochastic disturbance that is unknown. We wish to devise a data-driven algorithm that finds the optimal resource allocation even when the disturbance is initially unknown. We use stochastic gradient descent in order to minimize the cumulative loss function. In this section we explore the special case of square-loss with some assumptions on the structure of the disturbance. We derive a concrete algorithm and a corresponding bound for this special case. It is possible to easily extend this algorithm to various other scenarios. We make the following assumptions on the structure of the disturbance:

- The disturbance and data-points are independent (X_i is independent from δ_i).
- The disturbance is independent between features.
- The distribution of the disturbance in each feature is symmetric.
- The second moment of the disturbance, $\sigma_i^2(r_i)$ is convex in r_i .

The last assumption is reasonable since we expect diminishing return from increasing allocated resources. We use ridge regularization and bound the possible set of classifiers by $\|w\|_2 \leq B_w$.

The optimization problem can be stated as:

$$\begin{aligned} \min_{r,w} \quad & \sum_{t=1}^T l(w, r, x^t, y^t) \triangleq \sum_{t=1}^T \sum_{i=1}^d w_i^2 \sigma_i^2(r_i) + (w^T x^t - y^t)^2 \\ \text{s.t.} \quad & \sum_{i=1}^d r_i = R \\ & r_j \geq 0 \quad \forall j \end{aligned} \tag{8}$$

$$\|w\|_2 \leq B_w. \tag{9}$$

The gradient is given by:

$$\begin{aligned} \frac{\partial l}{\partial w_i} &= 2w_i \sigma_i^2(r_i) + 2x_i(w^T x - y) \\ \frac{\partial l}{\partial r_i} &= w_i^2 \frac{\partial \sigma_i^2(r_i)}{\partial r_i}. \end{aligned}$$

Since $\frac{\partial \sigma_i^2(r_i)}{\partial r_i}$ is unknown we will approximate it using the Kiefer-Wolfowitz procedure [11]. This results in $\frac{\partial \sigma_i^2(r_i)}{\partial r_i} \approx \frac{\sigma_i^2(r_i + \epsilon) - \sigma_i^2(r_i)}{\epsilon}$. We will denote by $\Pi(w, r)$ the projection of classifier w and resource vector r into the set of feasible solutions $\mathcal{N} \triangleq \{\|w\|_2 < B_W, \sum r_i = R, r_i > 0\}$. We further denote the maximum distance between two vectors in this set by $B = 2\sqrt{R^2 + B_W^2}$.

It is now possible to use standard stochastic gradient descent. Following the Kiefer-Wolfowitz procedure, at each step measure two data points with two slightly different resource allocations. Then, estimate the gradient and update the classifier and resource allocation accordingly. Finally, project the solution into the feasible solutions space and continue to the next step. The resulting algorithm is presented as Algorithm 1.

Algorithm 1 Learning when the disturbance is unknown

Parameters B_W, R, ϵ

initialize $w_1 = 0, r_i^1 = R/d$

for $t = 1, 2, 3, \dots, T$ **do**

 receive $\hat{x}^{t,1}, y^t$ using resource distribution r^t

 receive $\hat{x}^{t,2}, y^t$ using resource distribution $r^t + \epsilon$

$\eta = 1/\sqrt{t}$

for $i=1, \dots, d$ **do**

$w'_i = w_i^t - \eta((\langle w^t, \hat{x}^{t,1} \rangle - y^t) \hat{x}_i^{t,1})$

$r_i = r_i^t - \eta \frac{(w_i^t)^2 [(\hat{x}_i^{t,2})^2 - (\hat{x}_i^{t,1})^2]}{\epsilon}$

end for

$(w^{t+1}, r^{t+1}) = \Pi(w, r)$; $\Pi(w, r)$ is the projection into the feasible solutions space

end for

It is easy to verify that the estimated gradient is indeed unbiased. Notice that unlike standard on-line learning the measurement x_n are not i.i.d. since choosing r creates a coupling between measurements. However, the “noise” of the estimated gradient is a martingale difference sequence and therefore stochastic estimation theory can be easily applied.

We proceed to bound the regret which arise from algorithm 1. Since we use Keifer-Wolfowitz procedure the regret must be measured in comparison to the biased functions created by the procedure. Namely, $\tilde{\sigma}_i^2(r) = \int_0^r \frac{\sigma_i^2(s+\epsilon) - \sigma_i^2(s)}{\epsilon} ds$ and $\tilde{l}(w, r, x, y) \triangleq \sum_{i=1}^d w_i^2 \tilde{\sigma}_i^2(r_i) + (w^T x - y)^2$. When ϵ is small enough $\tilde{\sigma}_i^2(r)$ is approximately $\sigma_i^2(r)$. It is now possible to derive a bound on the regret.

Theorem 4. If $\tilde{l}(w, r, x, y)$ is jointly convex in w, r for every x, y , $\mathbb{E}(x) = 0$, $\mathbb{E}(\|x\|_2^2) = 1$, $\mathbb{E}(\|x\|_2^4) = B_x^4$, $\mathbb{E}((\hat{x}_i - x_i)^2) \leq B_\delta^2$ and $\mathbb{E}((\hat{x}_i - x_i)^4) \leq B_\delta^4$ then

$$\mathbb{E}\left(\sum_{i=1}^T (w^t x^{t,1} - y^t)^2\right) - \min_{(w,r) \in \mathcal{N}} \sum_{i=1}^T \tilde{l}(w, r, x^t, y^t) \leq \frac{B\sqrt{T}}{2} + \left(\sqrt{T} - \frac{1}{2}\right) \|\nabla l\|^2.$$

Where,

$$\begin{aligned}
B &= 2\sqrt{R^2 + B_W^2} \\
\|\nabla l\|^2 &= 2B_w^2 B_{\tilde{x}}^4 + 2B_{\tilde{x}}^2 + \frac{2B_{\tilde{x}}^4 B_W^4}{\epsilon^2} \\
B_{\tilde{x}}^4 &= B_x^4 + 6B_x^2 B_\delta^2 + B_\delta^4 \\
B_{\tilde{x}}^2 &= B_x^2 + B_\delta^2
\end{aligned} \tag{10}$$

The proof follows similar lines to that used to derive a bound in [4] and can be found in the appendix.

Theorem 4 implies that the optimal classifier and optimal resource allocation can be learned with sub-linear regret. Note that decreasing ϵ , which is the step-size used to estimate the gradient, will increase learning time. This is since we assume that noise is independent between samples. In this setting decreasing ϵ increases the noise level in estimating the gradient. Choosing large ϵ , however, can result in large bias from the optimal solution. The next two remarks show that assuming some dependence between samples may *reduce* learning time significantly.

Remark 1. The term $\frac{2B_{\tilde{x}}^4 B_W^4}{\epsilon^2}$ can be quite large. For reducing the variance in the learning process it is possible at some cases during training to sample multiple times the same data point. In such cases it is possible to derive a much better bound in which $\|\nabla l\|^2 = 2B_w^2 B_{\tilde{x}}^4 + 2B_{\tilde{x}}^2 + \frac{2B_\delta^4 B_W^4}{\epsilon^2}$.

Remark 2. In many cases the measurements noise of the same sample with different resources is correlated. This is for example the case when the resource is CPU time and the disturbance is caused from processing only part of the data. Two acquisitions of the same sample share a vast amount of common data. In such cases the difference between measurements with $r + \delta$ and r can be bounded much more tightly than the bound used in Theorem 4. If $\frac{(\hat{x}_i^{t,2} - x_i^t)^2 - (\hat{x}_i^{t,1} - x_i^t)^2}{\epsilon} \leq B_{grad}$ then $\|\nabla l\|^2$ in Theorem 4 can be rewritten as $\|\nabla l\|^2 = 2B_w^2 B_{\tilde{x}}^4 + 2B_{\tilde{x}}^2 + 2B_W^4 B_{grad}^2$.

Algorithm 2 Efficient learning from noisy data

Parameters $\eta, B_W, R, r(w)$
initialize $w_1 = 0, r_i^1 = R/d$
for $t = 1, 2, 3, \dots, T$ **do**
 receive x_t, y_t using resource distribution r^t
 $\nabla_t = 2(\langle w_t, x_t \rangle - y_t)x_t - \Sigma(r^t)w_t$
 $w' = w_t - \eta \nabla_t$
 $w_{t+1} = \arg \min_{|u|_1 < B_W} \|w' - u\|_2$
 $r^{t+1} = r(w_{t+1})$
end for

5 Learning from noisy data

In this section we explore the situation where the learning set is noisy while the test set is of perfect quality. This is the case in certain medical examinations where in the learning phase it is difficult to persuade a subject to go through extensive testing while at test time a patient suspected of having a serious disease will agree to such testing [4]. We adopt the framework in [4] that considered learning from noisy data. In our setting, however, the noise distribution can be controlled (under some resource constraints) by the learner. As we will show this control can produce a more efficient learning process. The on-line learning scheme fits this scenario since the optimal noise allocation depends on the classifier. We will focus our attention on the case of squared-loss. In [4] the authors develop an algorithm for online learning from noisy data. Their algorithm uses stochastic gradient descent in order to optimize the expected loss. Our algorithm is a modification of the one presented in [4] to include the control over resources. We will use lasso regularization in order to bound the set of classifiers, namely $|w|_1 < B_w$. The algorithm is presented as Algorithm 2. The algorithm receives as input the step size η , the lasso parameter B_w and some function which assign optimal resources for a known classifier $r(w)$. Examples for possible $r(w)$ had been given in section 2. The covariance matrix of the disturbance which results from using resources vector r is denoted by $\Sigma(r)$. Notice that $\Sigma(r)$ is diagonal and assumed known. We focus on the case where the disturbance is Gaussian with standard deviation $\sigma_i(r_i) = \frac{1}{\sqrt{r_i}}$.

Our results are based on the following lemma which is an adaptation of Theorem 2

from [4].

Lemma 2. Assume $\mathbb{E}_t(\|\nabla_t\|^2) \leq G$ and set $\eta = B_w/\sqrt{T}$ then the regret of Algorithm 2 satisfy $\mathbb{E}(\sum_{i=1}^T (w_i^T x_t - y_t)^2) - \min_{|w|_1 < B_W} (\sum_{i=1}^T (w^T x_t - y_t)^2) \leq \frac{1}{2}(G+1)B_W\sqrt{T}$.

Since the proof of this lemma is very similar to the one used to produce the results in [4] we refer the reader to [4].

We now move on to show that a proper choice of resources may improve learning. We assume the problem is normalized such that $\mathbb{E}(y) = 0$, $\mathbb{E}(y^2) = 1$, $\mathbb{E}(x) = 0$ and $\mathbb{E}(\|x\|_2^2) = 1$. We further denote $\mathbb{E}(\|x\|_2^4) = B_x^4$. The following two theorems show that proper allocation of resources can improve the efficiency of learning by $O(d)$. More specifically the regret will be $O(B_w^3 d^2, \sqrt{T})$ instead of $O(B_w^3 d^3, \sqrt{T})$.

Theorem 5. Assume $r_i^t(w) = \frac{R}{d}$ and $\eta = B_w/\sqrt{T}$. Then

$$\mathbb{E}(\sum_{i=1}^T (\langle w_t, x_t \rangle - y_t)^2) - \min_{|w|_1 < B_W} (\sum_{i=1}^T (\langle w, x_t \rangle - y_t)^2) \leq \frac{1}{2}(G+1)B_W\sqrt{T}.$$

Where,

$$G = 32B_w^2 \frac{d^3}{R^2} + 98B_w^2 \frac{d^2}{R^2} + 32B_W^2 \frac{d^2}{R} + 32B_W^2 \frac{d}{R} + 16 \frac{d^2}{R} + 32B_W^2 B_x^4 + 16 = O(B_W^2 d^3)$$

However, in case resources are allocated efficiently the corresponding bound is given by the following theorem.

Theorem 6. Assume $r_i^t(w) = \frac{R}{2d} + \frac{Rw_{ti}}{2|w_t|_1}$ and $\eta = B_w/\sqrt{T}$ then

$$\mathbb{E}(\sum_{i=1}^T (w_t x - y)^2) - \min_{|w|_1 < B_W} (\sum_{i=1}^T (\langle w_t, x \rangle - y)^2) \leq \frac{1}{2}(G+1)B_W\sqrt{T}$$

Where,

$$G = 64\frac{d^2}{R^2}B_W^2 + 64\frac{d^2}{R}B_W^2 + 32\frac{d^2}{R} + 392\frac{d}{R^2}B_W^2 + 64\frac{B_W^2}{R} + 32B_W^2B_x^4 + 16 = O(B_W^2d^2)$$

Notice that efficient learning requires some balance between two terms. The term $\frac{R}{2d}$ is required for estimating $\mathbb{E}(x)$ while the term $\frac{Rw_{ti}}{2|w_t|_1}$ is required for estimating $\mathbb{E}(w^Tx)$. We have created $r(w)$ by balancing those two terms evenly. It is possible that a different balance will provide better results.

Remark 3. When w is dense the efficient allocation is almost uniform. Therefore, the regret of the two resources allocation schemes should be similar. This is not evident from the bounds provided. The reason is that the proof of Theorem 5 uses the fact that in the worst case $|w|_2 = |w|_1$. In cases where w is dense this is loose. Using a tighter bound, $|w|_2 \cong \frac{|w|_1}{\sqrt{d}} \leq \frac{B_w}{\sqrt{d}}$ results in a bound with order $O(B_w^2d^2)$ for the uniform allocation case, similar to that received for efficient allocation of resources.

6 Simulation study

We tested the method on three datasets, one synthetic and two real-life problems from the UCI repository. Noise was added to all data artificially according to the relation $\sigma_i = \frac{1}{\sqrt{r_i}}$. For all datasets, measurement noise was created using the normal distribution with parameters $(0, \frac{\sigma_i}{3})$ and was added to the test samples. We applied the algorithm from the previous section to derive both an optimal classifier and an optimal resource allocation. The result given in Eq. (7) was used to derive the optimal resource allocation for a fixed classifier. We used hinge-loss as the loss function to be minimized and approximated $L(w, b, r)$ by using an adversarial ellipsoid uncertainty-set. Optimization was performed using the commercially available Mosek solver [1].

Synthetic problem. We generated 240000 samples uniformly distributed in a box in \mathbb{R}^3 . We used $z = x + 7y$ as the divider and created a data-set with labels that obey $sgn(z -$

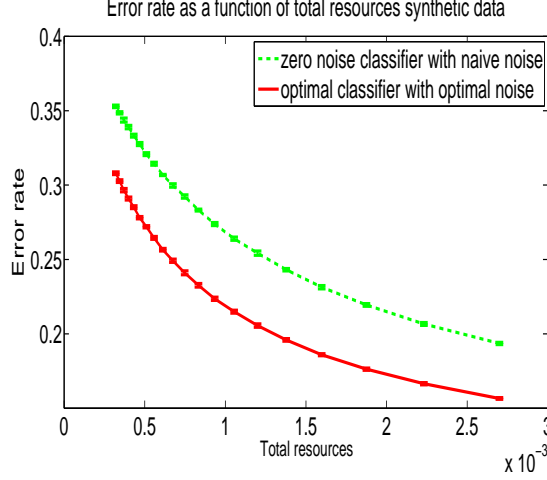


Figure 1: Error rate for synthetic data.

$x - 7y + N$), where N is some small Gaussian noise we added in order to make the data-set non-separable. A random subset of 10000 samples was used for learning while performance was measured on the rest. Tenfold cross validation was performed. The result for different R values is depicted at Figure 1. The method results in about 50% reduction in resources required for meeting the same error rate. In this case, the optimal classifier is similar to the classifier derived without noise and the benefit arise mainly from the redistribution of noise.

We wish to confirm the result of Theorem 2 using similar synthetic data-sets. For this purpose, we have generated nine data-sets each using as a divider $z = x + ay$ for $a = 1, 2, \dots, 9$. For each data-set we have extracted the resources needed for achieving an error rate of 0.15. We calculated the ratio between the total resources required when resources are allocated optimally and those required when resources are allocated uniformly. When $a = 1$ the optimal allocation is uniform and we expect no benefit (the ratio equals one). As we increase a , more resources should be allocated to y and therefore the ratio is improving (decreasing). Figure 2 shows the resulting graph compared with the theoretical result of Theorem 2 (using the optimal classifier). It can be seen that the simulation result is almost identical to the theoretical one, though contrary to the assumptions of Theorem 2 we are optimizing the hinge-loss and measuring error-rate. Observe in Figure 2 that considerable benefits arise even when the differentiation between features is rather small.

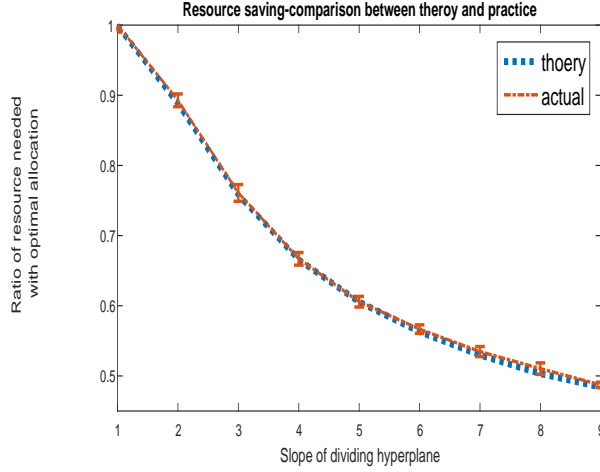


Figure 2: Ratio of resources needed for the same error level- synthetic data. Lower is better

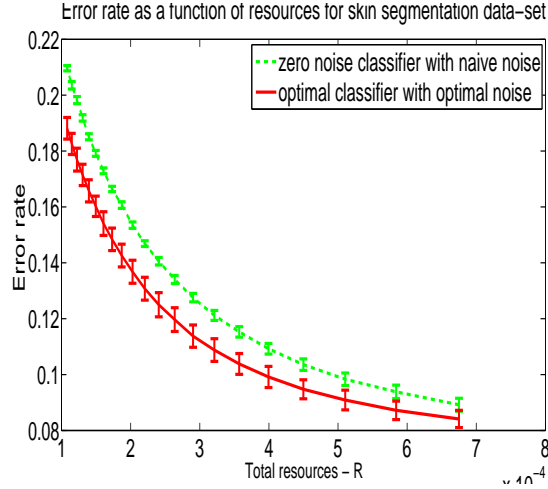


Figure 3: Error rate for skin segmentation data-set

Real data sets. Next, we tested the method on real-life databases from the UCI repository. We started with the skin segmentation data set [17] where RGB pixels are classified as skin or non-skin. Noise was added artificially to each pixel. From the 245057 available samples, a random subset of 10000 was used for learning while the rest was used to estimate performance. Ten-fold cross validation was performed. The results for different R values can be seen in Figure 3. It can be seen that the method results in about 30 % reduction in resources.

We tested the method on the breast cancer data set from the UCI repository [13]. This

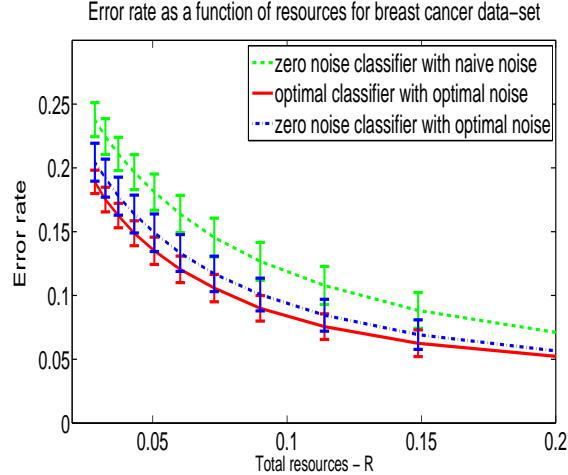


Figure 4: Error rate for breast cancer data-set

data-set contains 9 features that represent measurements from a biopsy and classified each sample as malignant or benign. The 683 samples were randomly divided, 2/3 of the data was used for training and the remaining 1/3 for testing. The results are depicted in Figure 4. The optimal classifier is different than the zero-noise classifier. In order to demonstrate what portion of the benefit arise from the resource allocation and what portion from the difference between the classifiers we added a plot of the error-rate of the zero-noise classifier when resources are allocated optimally. Most of the benefit comes from the correct allocation of resources.

7 Conclusion

We presented a method for optimal resource allocation in classification problems along with an analysis of the expected benefits from using this method. Our framework is general and we specialized it for the important special case of linear classifiers with Gaussian noise or with certain adversarial disturbances.

The framework we presented opens up several directions for future research. First, a natural extension of our work is to consider non-linear classifiers. This can be easily done using the “kernel-trick” computationally. However, while the disturbance (stochastic or

adversarial) has a comfortable shape in the input space, this does not necessarily happen in the feature space. This can probably be accommodated using the same techniques as [27] to obtain performance bounds.

Second, an expansion of the framework presented is the case where resources can be further divided between samples such that “hard” to classify examples will receive more resources. The key observation for this is the fact that allocation of resources between features is local in nature. The global cost function $L(h, r)$ can be replaced by $l(x, y, h, r)$ and therefore allows deciding on the allocation of resources for each sample separately. The optimal allocation creates a function $l(x, y, h, R)$ that can be used in the method presented in [18] to produce optimal allocation between samples.

Finally, the simulation results in this paper include only noise that was artificially generated. This is due to the complexity of creating a closed-loop system that controls the acquisition process. We believe that closing a complete feedback loop in applications such as sensor networks and radar will provide similar benefit to that presented as long as the noise is appropriately modelled.

8 Appendix

Proof of Theorem 1

Proof. We start by proving the following lemma:

Lemma 3. Let $L(w, b, \sigma)$ be an acceptable loss function. If $L(w, b, r)$ is twice differentiable in r then it is convex in r .

Proof. Since L is twice differentiable in r we can calculate the Hessian

$$\frac{\partial^2 G}{\partial r_i \partial r_j} = \frac{\partial^2 L}{\partial^2 \sigma} \frac{\partial \sigma}{\partial r_i} \frac{\partial \sigma}{\partial r_j} + \frac{\partial L}{\partial \sigma} \frac{\partial^2 \sigma}{\partial r_i \partial r_j}.$$

The first term is a positive semi-definite matrix that is multiplied by a positive factor

(since L is convex). The second term is the Hessian of σ which is positive semi-definite (since $\sigma(r)$ is convex) multiplied by a positive factor (since L is increasing). Therefore, the Hessian is positive semi-definite and L is convex in r .

□

We now continue to prove the theorem by noting that problem (1) can be rewritten as

$$\begin{aligned}
& \min_{w,b}(\min_r(L(w,b,r))) \\
& s.t \\
& \sum_{i=1}^d r_i = R \\
& r_i > 0
\end{aligned} \tag{11}$$

The inner optimization is convex, therefore necessary and sufficient conditions are given by Karush-Khun-Tucker

$$\begin{aligned}
\frac{\partial L}{\partial r_i} &= \frac{\partial L}{\partial \sigma} \frac{\partial \sigma}{\partial r_i} = -\lambda + \mu_i \\
\mu_i r_i &= 0 \\
\sum_{i=1}^d r_i &= R \\
\mu_i &\geq 0.
\end{aligned}$$

Since $\frac{\partial L}{\partial \sigma}(r)$ is positive and the same for each r_i we can denote $\tilde{\lambda} = \lambda(\frac{\partial L}{\partial \sigma})^{-1}$ and obtain the result. □

Proof of Theorem 2

Proof. From the definition of $R_{unif}(w, l)$ and $R_{opt}(w, l)$ it holds that $L(w, b, r_{unif}) = L(w, b, r_{opt}(w))$.

Now,

$$\frac{d|w|_2^2}{R_{unif}} + \mathbb{E}((wx - y)^2) = \frac{|w|_1^2}{R_{opt}} + \mathbb{E}((wx - y)^2)$$

Since the second term is the same in both sides of the equality it is easily derived that

$$\frac{R_{unif}(w, l)}{R_{opt}(w, l)} = \frac{d|w|_2^2}{|w|_1^2}.$$

□

Proof of Theorem 3

Proof. Using Lemma 1, problem (6) turns into:

$$\begin{aligned}
& \min_{\mathcal{N}_0} \min_{w,b,\xi} \sup_{\delta \in \mathcal{N}_0} (w^T \delta) + \sum_{i=1}^M \xi_i \\
& s.t : \\
& \xi_i \geq 1 - y_i(w^T x_i + b) \quad i = 1 \dots, M \\
& \xi_i \geq 0 \quad i = 1 \dots, M
\end{aligned} \tag{12}$$

substituting the order of the min prove the theorem. □

Proof of Theorem 4

Proof. We will first cite a slight adaptation of theorem 1 from [31] (similar adaptation was made in [4])

Lemma 4. Assume $\max_{t=1,\dots,T} \mathbb{E}(\|\nabla l(w^t, r^t)\|^2) \leq \|\nabla l\|^2$ then the regret of Algorithm 1 satisfies $\mathbb{E}(\sum_{i=1}^T (w_t x - y)^2 - \min_{|w|_1 < B_W} (\sum_{i=1}^T (\langle w_t, x \rangle - y)^2)) \leq \frac{B\sqrt{(T)}}{2} + (\sqrt{(T)} - \frac{1}{2})\|\nabla l\|^2$ where $B = 2\sqrt{R^2 + B_W^2}$

□

Now it is only left to prove that $\max_{t=1,\dots,T} \mathbb{E}(\|\nabla l(w^t, r^t)\|^2) \leq 2B_w^2 B_{\tilde{x}}^4 + 2B_{\tilde{x}}^2 + \frac{2B_{\tilde{x}}^4 B_W^4}{\epsilon^2}$

$$\begin{aligned}
\mathbb{E}(\|\nabla l(w^t, r^t)\|^2) &= \mathbb{E}[|(\langle w, \tilde{x} \rangle - y)\tilde{x}|^2 + \sum_{i=1}^d w_i^4 \frac{((\hat{x}_i^{t,2})^2 - (\hat{x}_i^{t,1})^2)^2}{\epsilon^2}] \\
&\leq 2\|w\|^2 \mathbb{E}(\|\tilde{x}\|^4) + 2\mathbb{E}(y^2 \|\tilde{x}\|^2) + \frac{\|w\|^4}{\epsilon^2} \max_{i=1,\dots,d} \mathbb{E}[(\hat{x}_i^{t,2})^2 - (\hat{x}_i^{t,1})^2] \\
&\leq 2B_w^2 B_{\tilde{x}}^4 + 2B_{\tilde{x}}^2 + \frac{2B_{\tilde{x}}^4 B_W^4}{\epsilon^2}.
\end{aligned}$$

The first inequality results from the fact that $||a + b||^2 \leq 2||a||^2 + 2||b||^2$. The second inequality stands since

$$\mathbb{E}(\tilde{x}^4) \leq B_x^4 + 6B_x^2 B_\delta^2 + B_\delta^4$$

$$\mathbb{E}(\tilde{x}^2) \leq B_x^2 + B_\delta^2$$

$$y^2 = 1.$$

Proof of Theorem 5

Proof. Denote the noisy measurement \tilde{x} as $x + N$ where N is the noise vector. The following relations are obtained by assigning $r_i = \frac{R}{d}$ and $\mathbb{E}||N_i||_2^2 = \frac{1}{r_i} = \frac{d}{R}$:

$$||\Sigma_t w_t||^2 = \sum_{i=1}^d \frac{w_i^2}{r_i^2} \leq \frac{d^2}{R^2} B_W^2$$

$$\mathbb{E}(||\langle W_t, N \rangle||_2^2) = \sum_{i=1}^d \frac{w_i^2}{r_i} \leq \frac{d}{R} B_W^2$$

$$\mathbb{E}(||N||_2^2) = \sum_{i=1}^d \frac{1}{r_i} \leq \frac{d^2}{R}.$$

Also,

$$\mathbb{E}(||\langle w, N \rangle||^2) = \left(\sum_{i=1}^d w_i N_i \right)^2 ||N||_2^2 = \sum_{i,j,k} w_i w_j N_i N_j N_k^2.$$

Since N_i is a zero mean gaussian random variable where for $i \neq j$ N_i and N_j are inde-

pendent all expectation of odd power in N_i is 0. in addition $\mathbb{E}(N_i^4) = 3\mathbb{E}(N_i^2)$. Now.

$$\begin{aligned}\mathbb{E}(\| \langle w, N \rangle \|^2) &= \sum_{i=1}^d w_i^2 E(N_i^2 \sum_{k=1}^d N_k^2) \\ &= \sum_{i=1}^d w_i^2 E(N_i^4) + \sum_{i,j,i \neq j}^d w_i^2 E(N_i^2 N_j^2) \\ &\leq 3 \frac{d^2}{R^2} B_W^2 + \frac{d^3}{R^2} B_W^2.\end{aligned}$$

Now, using the fact that $\|a + b\|^2 < 2\|a\|^2 + 2\|b\|^2$ at each stage,

$$\begin{aligned}\mathbb{E}(\|\nabla_t\|_2^2) &= E_t \|2(\langle w_t, \tilde{x}_t \rangle - y_t) \tilde{x}_t - \Sigma_t w_t\|_2^2 \\ &\leq 8\mathbb{E}(\|(\langle w_t, \tilde{x}_t \rangle - y_t) \tilde{x}_t\|^2) + 2\|\Sigma_t w_t\|^2 \\ &\leq 16\mathbb{E}(\|(\langle w_t, x_t \rangle + \langle w_t, N \rangle)(x_t + N)\|^2) + 16E(\|y_t(x_t + N)\|^2) + 2\|\Sigma_t w_t\|^2 \\ &\leq 32\mathbb{E}(\|(\langle w_t, x_t \rangle) x_t\|^2) + 32\mathbb{E}(\|(\langle w_t, N \rangle) N\|^2) \\ &\quad + 32\mathbb{E}(\|(\langle w_t, x_t \rangle) N\|^2) + 32\mathbb{E}(\|(\langle w_t, N \rangle) x_t\|^2) \\ &\quad + 16\mathbb{E}(\|y_t x_t\|^2) + 16E(\|y_t N\|^2) + 2\|\Sigma_t w_t\|^2 \\ &\leq 32B_w^2 \frac{d^3}{R^2} + 98B_w^2 \frac{d^2}{R^2} + 32B_W^2 \frac{d^2}{R} + 32B_W^2 \frac{d}{R} + 16 \frac{d^2}{R} + 32B_W^2 B_x^4 + 16 = G\end{aligned}$$

where the last inequality is due to the relations above.

□

Proof of Theorem 6

Proof. Now, $r_i(w) = \frac{R}{2d} + \frac{Rw_i}{2|w|_1}$. Therefore $r_i \geq \frac{R}{2d}$ and $r_i \geq \frac{Rw_i}{2|w|_1}$. This results in $\mathbb{E}\|N_i\|_2^2 \leq \frac{2d}{R}$ and $\mathbb{E}\|N_i\|_2^2 \leq \frac{2|w|_1}{Rw_i}$

Now,

$$\begin{aligned} \|\Sigma_t w_t\|^2 &= \sum_{i=1}^d \frac{w_i^2}{r_i^2} \leq \frac{4d}{R^2} B_W^2 \\ \mathbb{E}(\| < W_t, N > \|_2^2) &= \sum_{i=1}^d \frac{w_i^2}{r_i} \leq \frac{2}{R} B_W^2 \\ \mathbb{E}(\|N\|_2^2) &= \sum_{i=1}^d \frac{1}{r_i} \leq \frac{2d^2}{R} \end{aligned}$$

In a similar fashion to the derivation in the proof of Theorem 5 it results that

$$E(\| < w, N > N \|^2) = \sum_{i=1}^d w_i^2 E(N_i^4) + \sum_{i,j,i \neq j}^d w_i^2 E(N_i^2 N_j^2) \leq 12 \frac{d}{R^2} B_W^2 + 4 \frac{d^2}{R^2} B_W^2.$$

The rest of the proof is similar to that of Theorem 5 using the above relations instead of the corresponding relations in Theorem 5.

□

References

- [1] M. ApS. *The MOSEK optimization toolbox for MATLAB manual. Version 7.1 (Revision 28)*., 2015.
- [2] M. Athans. On the determination of optimal costly measurement strategies for linear stochastic systems. *Automatica*, 8(4):397–412, 1972.
- [3] D. Avitzour and S. R. Rogers. Optimal measurement scheduling for prediction and estimation. *Acoustics, Speech and Signal Processing, IEEE Transactions on*, 38(10):1733–1739, 1990.
- [4] N. Cesa-Bianchi, S. Shalev-Shwartz, and O. Shamir. Online learning of noisy data. *Information Theory, IEEE Transactions on*, 57(12):7907–7931, 2011.
- [5] T. Gao and D. Koller. Active classification based on value of classifier. In *Advances in Neural Information Processing Systems*, pages 1062–1070, 2011.
- [6] A. O. Hero and D. Cochran. Sensor management: Past, present, and future. *Sensors Journal, IEEE*, 11(12):3064–3075, 2011.
- [7] C.-W. Hsu, C.-C. Chang, C.-J. Lin, et al. Technical report, a practical guide to support vector classification, 2003.
- [8] K. Jenkins and D. A. Castanon. Adaptive sensor management for feature-based classification. In *Decision and Control (CDC), 2010 49th IEEE Conference on*, pages 522–527. IEEE, 2010.
- [9] S. M. Kakade, K. Sridharan, and A. Tewari. On the complexity of linear prediction: Risk bounds, margin bounds, and regularization. In *Advances in neural information processing systems*, pages 793–800, 2009.
- [10] R. Kohavi and G. H. John. Wrappers for feature subset selection. *Artificial Intelligence*, 97(12):273 – 324, 1997.

- [11] H. J. Kushner and G. G. Yin. Stochastic approximation algorithms and applications. 1997.
- [12] D. J. Lizotte, O. Madani, and R. Greiner. Budgeted learning of naive-bayes classifiers. In *Proceedings of the Nineteenth conference on Uncertainty in Artificial Intelligence*, pages 378–385. Morgan Kaufmann Publishers Inc., 2002.
- [13] O. L. Mangasarian, W. N. Street, and W. H. Wolberg. Breast cancer diagnosis and prognosis via linear programming. *Operations Research*, 43(4):570–577, 1995.
- [14] L. Meier III, J. Peschon, and R. Dressler. Optimal control of measurement subsystems. *Automatic Control, IEEE Transactions on*, 12(5):528–536, 1967.
- [15] P. Melville, M. Saar-Tsechansky, F. Provost, and R. Mooney. Active feature-value acquisition for classifier induction. In *Data Mining, 2004. ICDM’04. Fourth IEEE International Conference on*, pages 483–486. IEEE, 2004.
- [16] F. Nan, J. Wang, and V. Saligrama. Feature-budgeted random forest. *arXiv preprint arXiv:1502.05925*, 2015.
- [17] A. D. Rajen Bhatt. Skin segmentation dataset. UCI Machine Learning Repository.
- [18] O. Richman and S. Mannor. Dynamic sensing: Better classification under acquisition constraints. In *Proceedings of The 32st International Conference on Machine Learning*, 2015.
- [19] M. Shakeri, K. Pattipati, and D. Kleinman. Optimal measurement scheduling for state estimation. *Aerospace and Electronic Systems, IEEE Transactions on*, 31(2):716–729, 1995.
- [20] P. K. Shivaswamy, C. Bhattacharyya, and A. J. Smola. Second order cone programming approaches for handling missing and uncertain data. *The Journal of Machine Learning Research*, 7:1283–1314, 2006.
- [21] S. Stein and J. Jones. *Modern communication principles: with application to digital signaling*. McGraw-Hill, 1967.
- [22] I. Steinwart. Consistency of support vector machines and other regularized kernel classifiers. *Information Theory, IEEE Transactions on*, 51(1):128–142, 2005.
- [23] K. Trapeznikov, V. Saligrama, and D. Castañón. Multi-stage classifier design. *Machine learning*, 92(2-3):479–502, 2013.
- [24] V. N. Vapnik. *Statistical Learning Theory*. Wiley-Interscience, 1998.
- [25] J. Wintenby and V. Krishnamurthy. Hierarchical resource management in adaptive airborne surveillance radars. *Aerospace and Electronic Systems, IEEE Transactions on*, 42(2):401–420, 2006.
- [26] N. Xiong and P. Svensson. Multi-sensor management for information fusion: issues and approaches. *Information fusion*, 3(2):163–186, 2002.
- [27] H. Xu, C. Caramanis, and S. Mannor. Robustness and regularization of support vector machines. *The Journal of Machine Learning Research*, 10:1485–1510, 2009.
- [28] H. Xu, C. Caramanis, and S. Mannor. A distributional interpretation of robust optimization. In *Communication, Control, and Computing (Allerton), 2010 48th Annual Allerton Conference on*, pages 552–556. IEEE, 2010.

- [29] Z. Xu, O. Chapelle, and K. Q. Weinberger. The greedy miser: Learning under test-time budgets. In *Proceedings of the 29th International Conference on Machine Learning (ICML-12)*, pages 1175–1182, 2012.
- [30] Z. Xu, M. Kusner, M. Chen, and K. Q. Weinberger. Cost-sensitive tree of classifiers. In *Proceedings of the 30th International Conference on Machine Learning (ICML-13)*, pages 133–141, 2013.
- [31] M. Zinkevich. Online convex programming and generalized infinitesimal gradient ascent. In *Proceedings of the 20th International Conference on Machine Learning (ICML-03)*, pages 928–936, 2003.